A Novel Dynamic Identity based Authentication Scheme for Multi-Server Environment using Smart Cards

Chengbo Xu[1,2,*], Zhongtian Jia[3], Fengtong Wen[2] and Yan Ma[1]

[1]*Institute of Network Technology Research, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*School of Mathematical Sciences, University of Jinan, Jinan 250022, China*
[3]*Shandong Provincial Key Laboratory of Network Based Intelligent Computing, Jinan 250022, China*

[*]*Corresponding author, E-mail: cbqysy@gmail.com*

Abstract

Remote user authentication scheme with key agreement is a very practical mechanism to verify a remote user and then provide secure communication. Furthermore, many network environments have been becoming multi-server based due to the rapid growth of computer networks. Therefore, more and more researches have been focused on proposing smart card based remote authentication scheme with session key agreement for multi-server environment. Recently, Tsaur, Li and Lee (2012) proposed such a novel scheme which adopts a self-verified timestamp technique to help the smart card based authentication scheme not only effectively achieve password-authenticated key agreement but also avoid the difficulty of implementing clock synchronization in multi-server environments. They claimed that their scheme is against various attacks and more efficient. However, we observe that Tsaur-Li-Lee's scheme is still vulnerable to off-line password guessing attack, insider attack and malicious user attack. Besides, Tsaur-Li-Lee's scheme has no password change phase and also suffers from weaknesses of static identity and inefficiency in wrong password detection. In this paper, we propose an improved dynamic identity based scheme to eliminate all the security and efficiency weaknesses without decreasing other security performances.

Keywords: *Authentication, Dynamic identity, Multi-server, Smart card*

1. Introduction

Due to low cost, cryptographic capacity and portability of smart cards, the smart card based remote user authentication scheme with key agreement is a very practical mechanism to verify a remote user and then provide secure communication [3. 6, 7, 8, 12, 16].

However, many network environments have been becoming multi-server based with the rapid growth of computer networks. In these environments, those traditional single-server authentication schemes mentioned above cannot be well applied directly since users have to repetitively register at each involved remote servers and simultaneously remember numerous different identities and passwords. To overcome these problems, a serial schemes [1, 5, 11, 13, 17, 20, 21, 22] have been proposed. These multi-server authentication schemes can be divided into two types, *i.e.*, public-key based authentication and hash-based authentication. In 2000, Lee and Chang [11] firstly proposed a user authentication and key distribution scheme based

on RSA cryptosystem and hash functions. Later, Wu and Hsu [21] pointed out that Lee-Chang's scheme [11] is vulnerable to impersonation attack and proposed their scheme to resist the attack. In the same year, Yang *et al.*, [22] identify Wu-Hsu's scheme is still not secure and vulnerable to another impersonation attack. To remove the weakness, they improve Wu-Hsu's scheme. The improved scheme achieves user anonymity, user identification and key agreement. In parallel, Tsaur *et al.*, [20] proposed a password authentication scheme based on RSA cryptosystem and Lagrange interpolating polynomial for multi-server networks. Li *et al.*, [13] proposed a remote user authentication scheme based on an artificial neural network. Lin *et al.*, [17] proposed such scheme based on ElGamal digital signature protocol. However, these multi-server schemes commonly suffer from low efficiency since all of them are based on public key cryptosystems.

An efficient dimension to construct remote user authentication schemes for multi-server environment is based on hash function solely or combined with symmetric cryptosystem. In 2004, Juang [5] proposed an efficient multi-server user authentication and key agreement protocol based on hash function and symmetric key cryptosystem to improve the efficiency of Lin *et al.*, scheme [17]. However, Juang's scheme is vulnerable to stolen smart card attack. Besides, this scheme is not reparable. To remedy these weaknesses, Chang and Lee [1] proposed a novel remote authentication scheme, which is still not secure and was found vulnerable to insider attack, spoofing attack and register center spoofing attack.

All aforementioned schemes share a common feature that the user's identity is always static in transaction sessions. This feature gives attackers a chance to gather partial information about user's login request messages and further trace the different requests belonging to the same user. To remove this potential safety hazard, Liao and Wang [15] proposed a dynamic identity based remote user authentication scheme involving simple hash function to achieve user's anonymity. Moreover, this scheme provides a secure method to update the user's password off-line. In the same year, Hsiang and Shih [4] identified that Liao-Wang's scheme is susceptible to an insider attack, replay attack, stolen smart card attack, user masquerade attack, server spoofing attack and is not repairable. To remedy these flaws, Hsiang and Shih [4] proposed an improved scheme. Later, Lee-Lin-Chang [9] and Sood-Sarje-Singh [18] pointed out Hsiang-Shih's scheme is still not secure and vulnerable to replay attack, impersonation attack, stolen smart card attack, server spoofing attack and is not easily reparable. Then, they proposed their improvements respectively. Recently, Li *et al.*, [14] found that Sood-Sarje-Singh's scheme [18] is also susceptible to leak of verifier attack and stolen smart card attack. Furthermore, they improve the scheme to remedy those weaknesses.

In this paper, we analyse the scheme proposed recently by Tsaur, Li and Lee, and find that it not only has no password changing phase, but also suffers from static identity weakness and inefficiency in wrong password detecting. Then, we point out their scheme is vulnerable to malicious user attack. Furthermore, we propose an improved dynamic identity based scheme to remedy all above weaknesses in security and efficiency without decreasing other security performances.

The rest of this paper is organized as follows: in Section 2, we provide a brief review of Tsaur-Li-Lee's scheme [19]. Section 3 points out the security and efficiency weaknesses of Tsaur-Li-Lee's scheme. The proposed scheme and corresponding scheme analysis are presented in Sections 4 and 5 respectively. Finally, we conclude the paper in Section 6.

The notations used throughout this paper are summarized in Table 1.

Table 1. Notations

RC	The registration center
U_i	The i th user
S_j	The j th server
UID_i	The identity of the user U_i
$CUID_i$	The dynamic identity generated by the user U_i for authentication
SID_j	The identity of the server S_j
x	The master secret key maintained by RC
ω_j	The secret key shared between RC and S_j
PW_i	The password of the user U_i
E_T_{ij}	The service period of S_j for U_i
$E_s(\cdot)$	The encryption function with secret key s
$D_s(\cdot)$	The decryption function with secret key s
$h(\cdot)$	A secure one-way hash function
$\oplus$	The bitwise exclusive-or operation
$\|$	Message concatenation operation
v_i, μ_i	U_i 's secret information
v_{ij}	The secret key shared between U_i and S_j
M_{ij}	The authentication message for U_i to login in S_j
sk_k	The k th session key

2. Review of Tsaur-Li-Lee's Scheme

In this section, we briefly review the Tsaur-Li-Lee's scheme. Their scheme includes two phases: registration phase, log-in and session key agreement phase; and involves three entities: users, servers and registration center. RC selects the master key x. Each server S_j needs to register himself/herself with RC using the corresponding identity SID_j. In the registration phase, the registration center (RC) computes $w_j = h(x \| SID_j)$ and then submits it to S_j through a secure channel. The scheme is summarized in Figure 1.

2.1. Registration Phase

Suppose that user U_i can get service granted only from $S = \{S_1, S_2, \cdots, S_r\}$, and the service periods of these servers $S_1, S_2, \cdots, S_r$ for U_i are $E\_T_{i1}, E\_T_{i2}, \cdots, E\_T_{ir}$, respectively. When the user U_i wants to become a legal client to access the systems, U_i first chooses his/her identity UID_i and password PW_i, and then sends them to RC over a secure channel. After verifying the qualification, RC will perform the following steps:

Step 1: Compute $v_i = h(x+1, UID_i)$ and $\mu_i = v_i \oplus h(PW_i)$.

Step 2: Compute $v_{ij} = h(v_i, SID_j)$ shared between v_i and S_j for all $S_j \in S$.

Step 3: Calculate $A_{ij} = E_{w_j \oplus E\_T_{ij}}(v_{ij})$ for all $S_j \in S$.

Step 4: Store UID_i, μ_i, E_T_{ij} and A_{ij} into a smart card and issue this card to U_i.

2.2. Log-in and Session Key Agreement Phase

When the user U_i wants to login the server S_j, he/she inserts his/her smart card into a card reader and then keys in his/her password PW_i. The following steps are:

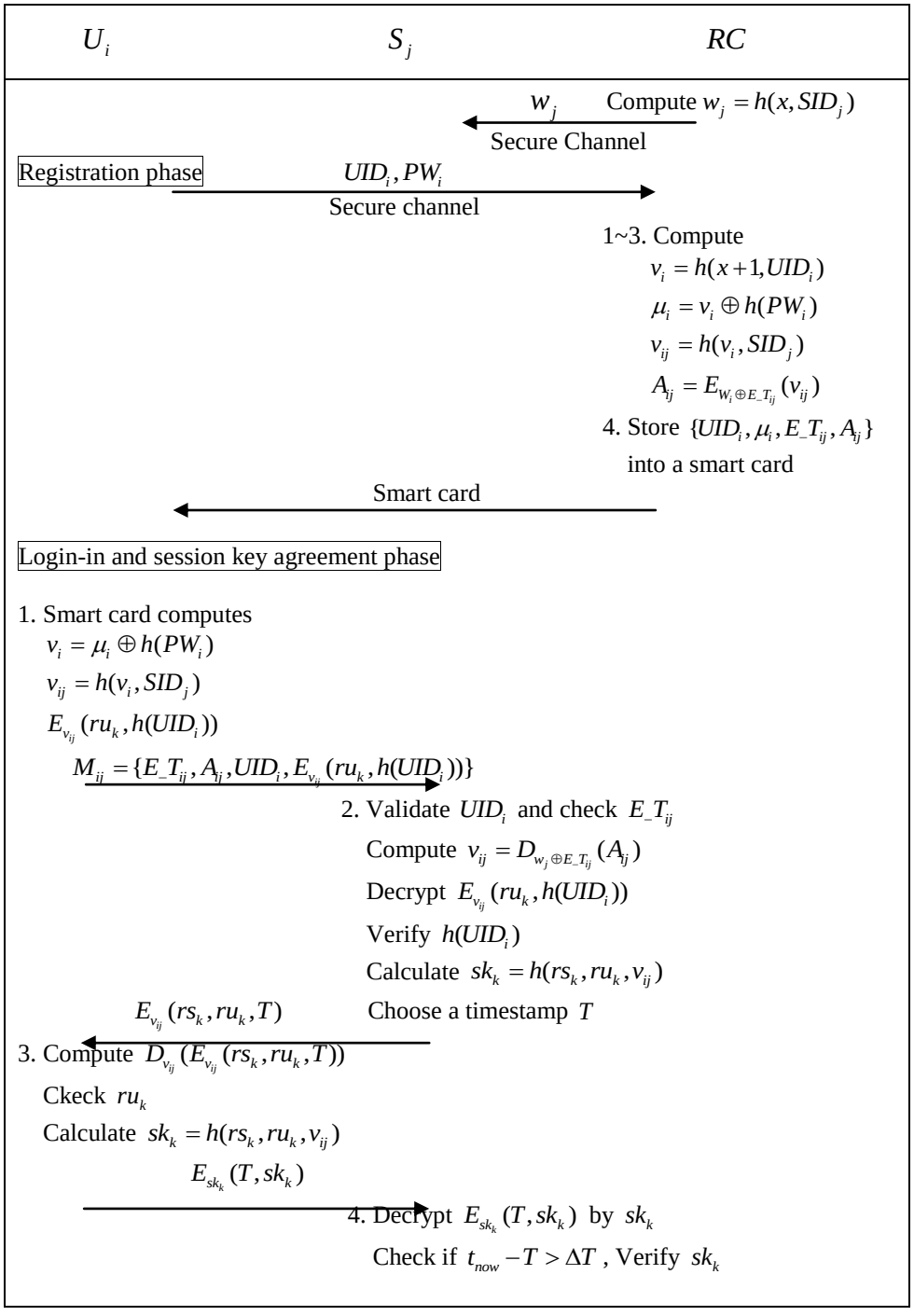

Figure 1. Tsaur-Li-Lee's Scheme

Step 1: The smart card computes $v_i = \mu_i \oplus h(PW_i)$ and $v_{ij} = h(v_i, SID_j)$, then chooses the k th random number ru_k and computes $E_{v_{ij}}(ru_k, h(UID_i))$, when U_i launches the k th log-in. Finally, the smart card constructs the message $M_{ij} = \{E\_T_{ij}, A_{ij}, UID_i, E_{v_{ij}}(ru_k, h(UID_i))\}$ and sends it to S_j.

Step 2: Upon receiving the message M_{ij}, S_j validates the format of UID_i. If it is invalid, S_j rejects the log-in request; Otherwise, the service period E_T_{ij} is further checked to see if it has expired. If not, S_j gets v_{ij} by decrypting A_{ij} with the secret key $w_j \oplus E\_T_{ij}$. Then S_j computes $D_{v_{ij}}(E_{v_{ij}}(ru_k, h(UID_i)))$ and obtains ru_k, $h(UID_i)$. If $h(UID_i)$ is not valid, S_j rejects the log-in request. Otherwise, S_j generates the k th random number rs_k, and computes the session key $sk_k = h(rs_k, ru_k, v_{ij})$. Then it sends $E_{v_{ij}}(rs_k, ru_k, T)$ to U_i, where T is the current timestamp.

Step 3: After receiving the message $E_{v_{ij}}(rs_k, ru_k, T)$, U_i 's smart card computes $D_{v_{ij}}(E_{v_{ij}}(rs_k, ru_k, T))$ and then checks the validity of ru_k. If it is valid, U_i 's smart card calculates the session key $sk_k = h(rs_k, ru_k, v_{ij})$ and the message $E_{sk_k}(T, sk_k)$ which is sent to S_j; Otherwise, U_i terminates this session.

Step 4: S_j decrypts the receiving message $E_{sk_k}(T, sk_k)$ with the secret key sk_k, and then checks whether $t_{now} - T > \Delta T$, where t_{now} represents S_j 's current time, and ΔT is the expected valid time interval for transmission delay. If the inequality is valid, S_j aborts the connection. Otherwise, it further checks the validity of sk_k derived from decrypting the message $E_{sk_k}(T, sk_k)$. If it is valid, the mutual authentication succeeds and the common session key sk_k is negotiated successfully. Otherwise, S_j terminates the session.

3. Weaknesses of Tsaur-Li-Lee's Scheme

In this section, we will show that Tsar-Li-Lee's scheme is vulnerable to off-line password guessing attack, privileged insider attack and malicious user attack. Among them, the first two attacks were also pointed out by Yitao Chen [2]. Besides, their scheme has the weaknesses of static user's identity and low efficiency in wrong password detection.

3.1. Off-line Password Guessing Attack

As pointed out by Messerges *et al.*, [10], the confidential information stored in a smart card could be extracted by physically monitoring its power consumption. Therefore, we assume attackers have the ability to extract the information stored in smart card.

Suppose the user U_i 's smart card is lost or stolen, and obtained by an attacker A. The attacker A can extract the values UID_i, μ_i, E_T_{ij} and A_{ij}, where $v_i = h(x+1, UID_i)$, $\mu_i = v_i \oplus h(PW_i)$, $v_{ij} = h(v_i, SID_j)$ and $A_{ij} = E_{w_j \oplus E\_T_{ij}}(v_{ij})$. With these information and an eavesdropped previously valid login message $M_{ij} = \{E\_T_{ij}, A_{ij}, UID_i, E_{v_{ij}}(ru_k, h(UID_i))\}$, the

attacker A can off-line guess the password as follows: 1) A selects a possible password PW_i^*, computes $v_i^* = v_i \oplus h(PW_i^*)$ and $v_{ij}^* = h(v_i^*, SID_j)$; 2) A decrypts $E_{v_{ij}}(ru_k, h(UID_i))$ with v_{ij}^*, and obtains ru_k^*, $h(UID_i)^*$; 3) A checks whether $h(UID_i)^*$ and $h(UID_i)$ are equal or not. If they are equal, A finds the correct password; Otherwise, A repeats 1)-3) until finding the correct password. After knowing PW_i, A can correctly compute $v_i = \mu_i \oplus h(PW_i)$ and $v_{ij} = h(v_i, SID_j)$. With the correct value v_{ij}, the attacker A can masquerade as the user U_i to login the server S_j or masquerade as S_j to fool U_i easily.

3.2. Privileged Insider Attack

In the Tsaur-Li-Lee's scheme, U_i sends directly the password PW_i and ID_i to RC when he/she wants to register himself/herself. If the system manager or a privileged insider A of the register center RC records these values, he/she could masquerade as U_i not only to login the servers in this system but also to access other system's servers, since many users commonly use the same password to access different applications or servers for their convenience of remembering the password and ease-of -use whenever required in real environment. Therefore, Tsaur-Li-Lee's scheme is vulnerable to privileged insider attack.

3.3. Malicious User Attack

A malicious privileged user U_i with knowledge ID_i and PW_i also can extract the information UID_i, μ_i, E_T_{ij} and A_{ij} stored in his/her smart card. Then he/she can masquerade as other user U_m to login into any server in the system as follows: 1) the malicious user U_i firstly computes $v_i = \mu_i \oplus h(PW_i)$ and $v_{ij} = h(v_i, SID_j)$; 2) U_i generates a nonce $ru_k^{'}$ and computes $E_{v_{ij}}(ru_k^{'}, h(UID_m))$; 3) U_i sends $M_{mj} = \{E\_T_{ij}, A_{ij}, UID_m, E_{v_{ij}}(ru_k^{'}, h(UID_m))\}$ to the server S_j. It can be easily seen that the forged login request message M_{mj} can pass the verification of S_j. Then S_j sends back the message $E_{v_{ij}}(rs_k, ru_k^{'}, T)$. When receiving the message, U_i with the knowledge v_{ij} can correctly decrypt it. Then he/she computes $sk_k = h(rs_k, ru_k^{'}, v_{ij})$, $E_{sk_k}(T, sk_k)$ and sends $E_{sk_k}(T, sk_k)$ to server S_j. The message $E_{sk_k}(T, sk_k)$ can easily pass the last verification of S_j. As such, the malicious user U_i successfully masquerades as user U_m to login into server S_j.

3.4. Low Efficiency in Wrong Password Detection

If the legal user U_i inputs a wrong password by mistake, this wrong password will not be detected until the remote server verifies $h(UID_i)$ in step 2 of the log-in and session key agreement phase. Therefore, Tsaur-Li-Lee's scheme is low efficient to detect the user's wrong password.

3.5. No password Change Phase

In Tsaur-Li-Lee's scheme, there is no password change phase. Actually, it is not difficult to add this phase. When the user U_i wants to change his/her password, he/she inserts the smart

card into a card reader, inputs the identity UID_i and password PW_i, then calls for changing password. U_i will select and input a new password PW_{new}. And then the smart card computes $\mu_{new} = \mu_i \oplus h(PW_i) \oplus h(PW_{new})$, and replaces μ_i with μ_{new}. As such, the password is changed.

However, since no wrong password detection mechanism is designed in the smart card, the password change phase would suffer from the following weakness. If an attacker A stole user U_i's smart card for a short time, he/she inserts U_i's smart card into a card reader, enters the UID_i and an arbitrary password PW_a, and calls for changing password. Then A enters an arbitrary new password PW_a^*. The smart card will compute $\mu_a = \mu_i \oplus h(PW_a) \oplus h(PW_a^*)$, which yields $v_i \oplus h(PW_i) \oplus h(PW_a) \oplus h(PW_a^*)$, and then replaces μ_i with μ_a without any checking. Later, the legal user U_i's succeeding login requests will be denied unless he/she re-registers with RC.

3.6. Weakness of Static User's Identity

Since the user's identity UID_i is static and transported in complete plaintext in Tsaur-Li-Lee's scheme. The user U_i would be traced and vulnerable to ID-theft attack as pointed out by Das, Saxena and Gulati (2004).

4. Our Proposed Scheme

In this section, we propose a dynamic identity based remote user authentication scheme for multi-server environment. The proposed scheme is the improvement of Tsaur-Li-Lee's scheme. It is free from all the attacks and weaknesses considered above. There are also three entities in our scheme, i.e. the user (U_i), the server (S_j) and the registration center (RC). RC is assumed to be trusted and responsible for registration of the U_i and S_j. RC selects the master key x which only it itself knows. Then it computes the key $w_j = h(x \| SID_j)$ and shares w_j with S_j through a secure channel. The proposed scheme involves four phases: registration phase, login phase, authentication and session key agreement phase, password change phase. The first three phases are summarized in Figure 2.

4.1. Registration Phase

Suppose that user U_i can get service granted only from $S = \{S_1, S_2, \cdots, S_r\}$, and the service periods of these servers $S_1, S_2, \cdots, S_r$ for U_i are $E\_T_{i1}, E\_T_{i2}, \cdots, E\_T_{ir}$, respectively. When the user U_i wants to register himself/herself with RC, U_i first selects his/her identity UID_i, password PW_i and generates a random number b_i. The steps of the registration are as follows:

Step R1. U_i computes $h(UID_i \| b_i)$ and sends it to RC over a secure channel.

Step R2. Upon receiving $h(UID_i \| b_i)$, RC computes U_i's secret information $v_i = h(x+1 \| h(UID_i \| b_i))$.

Step R3. RC computes $v_{ij} = h(v_i \| SID_j)$ and $A_{ij} = E_{w_j \oplus E\_T_{ij}}(v_{ij})$ for all $S_j \in S$.

Step R4. RC stores v_i E_T_{ij}, A_{ij} into a smart card and issues this card to U_i via a secure channel.

Step R5. After receiving the smart card, U_i inserts it into a dedicated card reader and inputs his/her UID_i and PW_i.

Step R6. The smart card computes $R_i = h(PW_i \| UID_i)$ and $\mu_i = v_i \oplus h(PW_i \oplus UID_i)$. Then it stores R_i and substitutes v_i with μ_i. Eventually, the smart card contains $\{R_i, \mu_i, E\_T_{ij}, A_{ij}, h(\cdot)\}$.

4.2. Login Phase

When the user U_i wants to access the resources of the server S_j. The steps are as follows:

Step L1. U_i inserts his/her smart card into a smart card reader and inputs the identity $UID_i^{'}$ and password $PW_i^{'}$. Then the smart card computes $R_i^{'} = h(PW_i^{'} \| UID_i^{'})$ and checks whether $R_i^{'} = R_i$ or not. If they are equal, it means U_i is a legal user; Otherwise, the smart card rejects this login request.

Step L2. After verification, the smart card computes $v_i = \mu_i \oplus h(PW_i \oplus UID_i)$, $v_{ij} = h(v_i \| SID_j)$. Then it chooses a random number ru_k and computes $CUID_i = E_{v_{ij}}(ru_k \| h(UID_i \oplus v_{ij}) \| UID_i)$, when U_i launches the k th log-in.

Step L3. The smart card constructs the login request message $M_{ij} = \{E\_T_{ij}, A_{ij}, CUID_i\}$ and sends it to the server S_j.

4.3. Authentication and Session Key Agreement Phase

Upon receiving the login request message $M_{ij} = \{E\_T_{ij}, A_{ij}, CUID_i\}$, the server S_j and the user U_i verify each other with the following steps:

Step V1. S_j computes $v_{ij} = E_{w_j \oplus E\_T_{ij}}(A_{ij})$, $D_{v_{ij}}(CUID_i)$ and obtains $ru_k^{'}$, $UID_i^{'}$, $h(UID_i \oplus v_{ij})^{'}$.

Step V2. S_j checks the format of $UID_i^{'}$. If it is invalid, S_j rejects the login request; Otherwise, S_j verifies whether $h(UID_i^{'} \oplus v_{ij}) = h(UID_i \oplus v_{ij})^{'}$ or not. If they are not equal, S_j rejects the login request; Otherwise, S_j further checks E_T_{ij} to see if it has expired. If E_T_{ij} has expired, S_j will terminate the service to U_i; Otherwise, S_j has authenticated U_i and processes the next step.

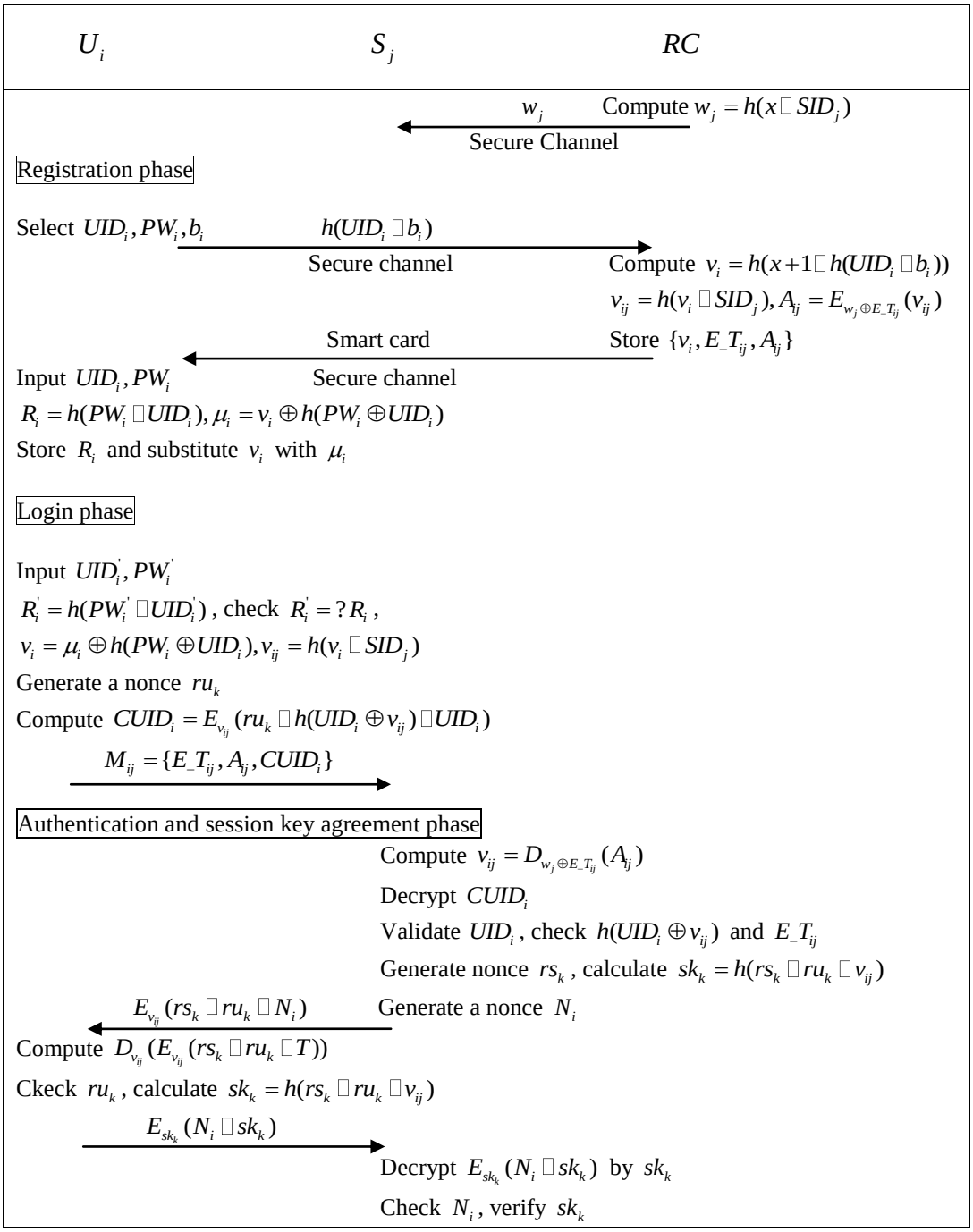

Figure 2. The Proposed Scheme

Step V3. S_j generates two random number rs_k, N_i, and calculates $sk_k = h(rs_k \| ru_k \| v_{ij})$.

Step V4. S_j computes $E_{v_{ij}}(rs_k \| ru_k' \| N_i)$ and then sends it to U_i.

Step V5. Upon receiving the message $E_{v_{ij}}(rs_k \| ru_k^{'} \| N_i)$, U_i decrypts it with the secret key v_{ij} and checks whether $ru_k = ru_k^{'}$. If they are equal, the validity of the server S_j is verified by U_i; Otherwise, U_i terminates the session.

Step V6. U_i computes $sk_k = h(rs_k \| ru_k \| v_{ij})$ and $E_{sk_k}(N_i \| sk_k)$. Then he/she sends $E_{sk_k}(N_i \| sk_k)$ to S_j.

Step V7. When receiving the message $E_{sk_k}(N_i \| sk_k)$, S_j decrypts it with the secret key sk_k and obtains $N_i^{'}$, $sk_k^{'}$. Then S_j checks whether $N_i^{'} = N_i$ or not. If they are not equal, S_j terminates the session; Otherwise, S_j further checks whether $sk_k^{'} = sk_k$ or not. If they are equal, the validity of the user U_i is verified; Otherwise, S_j terminates the session.

4.4. Authentication and Session Key Agreement Phase

This phase is invoked whenever U_i wants to change his/her password PW_i without the help of RC. The steps are as follows:

Step P1. U_i inserts his smart card into a smart card reader, then enters ID_i^*, PW_i^* and requests to change password.

Step P2. U_i's smart card computes $R_i^* = h(PW_i^* \| UID_i^*)$ and checks whether R_i^* and R_i are equal or not. If not, the smart card rejects the password change request; Otherwise, U_i selects a new password PW_{new} and inputs it.

Step P3. U_i's smart card computes $R_{new} = h(PW_{new} \| UID_i)$, $\mu_{new} = \mu_i \oplus h(PW_i \oplus UID_i) \oplus h(PW_{new} \oplus UID_i)$, and substitutes R_i, μ_i with R_{new}, μ_{new} respectively.

5. Security Analysis

In this section, we will mainly discuss the enhanced security and efficiency features of our improved scheme. The others are the same as Tsaur-Li-Lee's scheme.

5.1. User's Anonymity

The proposed scheme is a dynamic identity based scheme. In all of the four phases, the real identity UID_i is never transported in plaintext and cannot be computed by the attacker, even if he/she might stole the user's smart card and extract the information stored in the card, or intercept a previously valid login message. Concretely, the secure channel and the random number b_i are used to protect the user's identity from disclosure in the registration phase of our proposed scheme. In the login phase, the user U_i submits the masked identity $CUID_i = E_{v_{ij}}(ru_k \| h(UID_i \oplus v_{ij}) \| UID_i)$ rather than the real identity UID_i in the login request message. In the authentication and session key agreement phase, the real identity UID_i can be recovered only by server S_j since only server S_j can compute the secret key v_{ij} which will be used to decrypt the masked dynamic identity $CUID_i$. Based on the above analysis, we can see that the proposed scheme provides the user's anonymity.

5.2. Resist off-line Dictionary Attack

The off-line dictionary attack means that the attacker collects related information by various methods and then attempts to guess user U_i's identity UID_i or password PW_i using these information. As pointed out by Sood, Sarje and Singh [18], we also assume that it is impossible to guess the two parameters correctly at the same time in real polynomial time. However, generally speaking, an attacker has the capability to guess UID_i or PW_i individually due to the low entropy of UID_i and PW_i selected freely by user U_i himself/herself. In the proposed scheme, an attacker might collect the values $R_i = h(PW_i \| UID_i)$, $\mu_i = v_i \oplus h(PW_i \oplus UID_i)$, E_T_{ij} , $A_{ij} = E_{w_j \oplus E\_T_{ij}}(v_{ij})$ and $CUID_i = E_{v_{ij}}(ru_k \| h(UID_i \oplus v_{ij}) \| UID_i)$ through various methods, such as extracting the information stored in the stolen smart card or intercepting previously valid login request messages. According to the assumption above, the attacker cannot guess UID_i or PW_i from R_i. Moreover, he/she also cannot guess them from μ_i , E_T_{ij} , A_{ij} and $CUID_i$ without the knowledge v_i and w_i. Therefore, our proposed scheme resists the off-line dictionary attack.

5.3. Resist Insider Attack

In the registration phase of our proposed scheme, only the user U_i's knowledge $h(UID_i \| b_i)$ is sent to RC for registration. After receiving the smart card, the password PW_i is entered into the smart card to compute the wrong password detection message R_i and hider the high secret value v_i by the user himself/herself. So the system manager or a privileged insider A of the register center RC has no way to record the identity UID_i and password PW_i to initiate an insider attack. Even if A records the value $h(UID_i \| b_i)$, he/she cannot off-line guess the real identity UID_i without the random number b_i which is generated by U_i and removed immediately after the value $h(UID_i \| b_i)$ was computed. From above analysis, we can say the proposed scheme resist insider attack.

5.4. Resist Malicious User Attack

A malicious privileged user U_m with knowledge UID_m and PW_m can extract the information R_m, μ_m, E_T_{mj}, A_{mj} stored in his/her own smart card. U_m also can intercept or eavesdrop other user U_i's login request message $M_{ij} = \{E\_T_{ij}, A_{ij}, CUID_i\}$. In our proposed scheme, even if a malicious privileged user obtain all possible values mentioned above, he/she can not construct another valid login request message $M_{ij}^{'}$ to masquerade as U_i to login S_j, since he/she has no way to compute another $CUID_i$ to pass the check process of $h(UID_i \oplus v_{ij})$ without the secret key v_{ij}. Besides, U_m also cannot succeed in replay attack because he/she is unable to obtain the nonce N_i generated by S_j. Therefore, the proposed scheme can resist the malicious user attack.

5.5. Efficiency Improvement in Wrong Password Detection

If the user U_i inputs a wrong password $PW_i^{'}$ by mistake, this wrong password will be quickly detected by U_i's smart card since the smart card can check $h(PW_i^{'} \oplus UID_i)$ with the stored value R_i in step L1 of the login phase.

5.6. Cost and Functionality Analysis

In this subsection, we evaluate the computation cost and functionality of our proposed scheme through comparing with several related schemes. To analyze the computational complexity of these schemes, we define the notation T_h and T_s as the time complexity for hash function and symmetric cryptosystem respectively. The computation cost of exclusion-or and concatenation operations are usually neglected, since the two operations require very few computation.

We compare the cost of our proposed scheme and those four related schemes in Table 2. Since login and authentication phases should be implemented for each session, we mainly consider the computation cost of these two phases as shown in almost performance analysis of related works. Among the five schemes, the first three schemes are all based on hash function and symmetric cryptosystem. From Table 2, we can see that our proposed scheme needs almost the same cost as well as Tsaur-Li-Lee's scheme and Juang's scheme. Of course, it is worth two more hash operations to achieve these security and functionality features vis-a-vis Tsaur-Li-Lee's scheme.

Table 2. Cost Comparisons of our Scheme and Previously Proposed Schemes

	Ours	Tsaur-Li-Lee's scheme(2012)	Juang's scheme(2004)	Li *et al.*'s scheme(2011)	Tsai's scheme(2008)
Cost of user registration	$5T_h, 1T_s$	$3T_h, 1T_s$	$1T_h$	$1T_h$	$2T_h$
Cost of server registration	$1T_h$	$1T_h$	$1T_h$	$5T_h$	$1T_h$
Cost of login and verification					
User	$5T_h, 3T_s$	$2T_h, 3T_s$	$11T_h$	$3T_h$	$5T_h, 3T_s$
Server	$2T_h, 3T_s$	$2T_h, 3T_s$	$3T_h, 4T_s$	$5T_h$	$3T_h$
RC	$0T_h, 0T_s$	$0T_h, 0T_s$	$1T_h, 2T_s$	$14T_h$	$1T_h$

Table 3 lists the functionality comparison among those five schemes. It can be clearly seen that our scheme is more secure against various attacks than other four related schemes. Besides, our scheme and Tsaur-Li-Lee's scheme share a common feature that the *RC* does not play part in the user authentication process. Therefore, the cost of communication is more lower than other three scheme.

Table 3. Functionality Comparisons of our Scheme and Previously Proposed Schemes

	Ours	Tsaur-Li-Lee(2012)	Li *et al.*, (2011)	Tsai (2008)	Juang(2008)
No verification table	Yes	Yes	Yes	Yes	Yes
Computation cost	Low	Low	Low	Low	Low
Service period management	Yes	Yes	No	No	No
Single registration	Yes	Yes	Yes	Yes	Yes
No clock synchronization	Yes	Yes	Yes	Yes	Yes
Resist man-in-the middle attack	Yes	Yes	Yes	No	No
User's anonymity	Yes	No	Yes	No	No
Password change phase	Yes	No	Yes	Yes	Yes
Resist off-line dictionary attack	Yes	No	No	Yes	Yes
Resist insider attack	Yes	No	No	No	No
Resist malicious user attack	Yes	No	Yes	Yes	Yes
Efficiency in wrong password detection	Yes	No	Yes	Yes	Yes
No need for assistance of RC in authentication phase	Yes	Yes	No	No	No

6. Conclusions

In this paper, we have shown that Tsaur-Li-Lee's scheme is vulnerable to off-line password guessing attack, insider attack and malicious user attack. Besides, Tsaur-Li-Lee's scheme has no password change phase and also suffers from weaknesses of static identity and inefficiency in wrong password detection. Then we proposed a novel dynamic identity based scheme to eliminate the aforementioned weaknesses of Tsaur-Li-Lee's scheme. Through comparing with several related schemes, we demonstrated that the proposed scheme is more secure and efficient. Therefore, the proposed scheme is more practicable.

Acknowledgements

This work was partially supported by the Doctoral Fund of University of Jinan (Granted No. XBS0835), and the project of Jinan City Science and Technology Program (Granted No. 201202014).

References

[1] C. C. Chang and J. S. Lee, "An efficient and secure multi-server password authentication scheme using smart cards", Proceedings of the third international conference on cyberworlds, **(2004)**, pp. 417-22.
[2] Y. T. Chen, "Comments of an efficient and secure multi-server authentication scheme with key agreement", eprint.iacr.org, vol. 702, **(2011)**.
[3] C. I. Fan, Y. C. Chan and Z. K. Zhang, "Robust remote authentication scheme with smart cards", Computers & Security, vol. 24, **(2005)**, pp. 619-28.
[4] H. C. Hsiang and W. K. Shih, "Improvement of the secure dynamic ID based remote user authentication scheme for multi-server environment", Computer Standards & Interfaces, vol. 31, **(2009)**, pp. 1118-1123.
[5] W. S. Juang, "Efficient multi-server password authenticated key agreement using smart cards", IEEE Transaction on Consumer Electronics, vol. 50, **(2004)**, pp. 251-255.
[6] J. Y. Kim, H. K. Choi and J. A. Copeland, "Further improved remote user authentication scheme", IEICE Transaction on Fundamentals, E94-A, **(2011)**, pp. 1426-1433.
[7] S. K. Kim and M. G. Chung, "More secure remote user authentication scheme", Computer Communications, vol. 32, **(2009)**, pp. 1018-1021.

[8] W. C. Ku and S. M. Chen, "Weakness and improvements of an efficient password based remote user authentication scheme using smart cards", IEEE Transaction on Consumer Electronics, vol. 50, **(2004)**, pp. 204-207.
[9] C. C. Lee, T. H. Lin and R. X. Chang, "A secure dynamic ID based remote user authentication scheme for multi-server environment using smart cards", Expert Systems with Applications, vol. 38, **(2011)**, pp. 13863-13870.
[10] T. S. Messerges, E. A. Dabbish and R. H. Sloan, "Examining smart-card security under the threat of power analysis attacks", IEEE Transaction on Computers, vol. 51, **(2002)**, pp. 541-552.
[11] W. B. Lee and C. C. Chang, "User identification and key distribution maintaining anonymity for distributed computer network", International Journal of Computer Systems Science & Engineering, vol. 15, **(2000)**, pp. 211-214.
[12] C. T. Li and M. S. Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards", Journal of Network and Computer Applications, vol. 33, **(2010)**, pp. 1-5.
[13] L. H. Li, L. C. Lin and M. S. Hwang, "A remote password authentication scheme for multi-server architecture using neural networks", IEEE Transactions on Neural Networks, vol. 12, **(2001)**, pp. 1498-1504.
[14] X. Li, Y. P. Xiong, J. Ma and W. D. Wang, "An efficient and security dynamic identity based authentication protocol for multi-server architecture using smart cards", Journal of Network and Computer Applications, vol. 35, **(2012)**, pp. 763-769.
[15] Y. P. Liao and S. S. Wang, "A secure dynamic ID based remote user authentication scheme for multi-server environment", Computer Standards & Interfaces, vol. 31, **(2009)**, pp. 24-29.
[16] J. Y. Liu, A. M. Zhou and M. X. Gao, "A new mutual authentication scheme based on nonce and smart cards", Computer Communications, vol. 31, **(2008)**, pp. 2205-2209.
[17] I. C. Lin, M. S. Hwang and L. H. Li, "A new remote user authentication scheme for multi-server architecture", Future Generation Computer Systems, vol. 1, **(2003)**, pp. 13-22.
[18] S. K. Sood, A. K. Sarje and K. Singh, "A secure dynamic identity based authentication protocol for multi-server architecture", Journal of Network and Computer Applications, vol. 34, **(2011)**, pp. 609-618.
[19] W. J. Tsaur, J. H. Li and W. B. Lee, "An efficient and secure multi-server authentication scheme with key agreement", The Journal of System and Software, vol. 85, **(2012)**, pp. 876-882.
[20] W. J. Tsaur, C. C. Wu and W. B. Lee, "A flexible user authentication for multi-server Internet Services", Networking-JCN2001LNCS, 2093, Springer-Verlag, **(2001)**, pp. 174-183.
[21] T. S. Wu and C. L. Hsu, "Efficient user identification scheme with key distribution preserving anonymity for distributed computer networks", Computer & Security, vol. 23, **(2004)**, pp. 120-125.
[22] Y. Yang, S. Wang, F. Bao, J. Wang and R. Deng, "New efficient user identification and key distribution scheme providing enhanced security", Computer & Security, vol. 23, **(2004)**, pp. 697-704.